



Lab-26: Docker & Elastic Setup

Tatiana Meneses

January 20th, 2025

Overview

This exercise is designed to guide you through installing Docker and Elastic on an Ubuntu machine, followed by verifying network activity through **ping** and **nmap** scans.

This hands-on activity will introduce you to setting up an Elastic SIEM environment, which will be further demonstrated during class.

Objectives

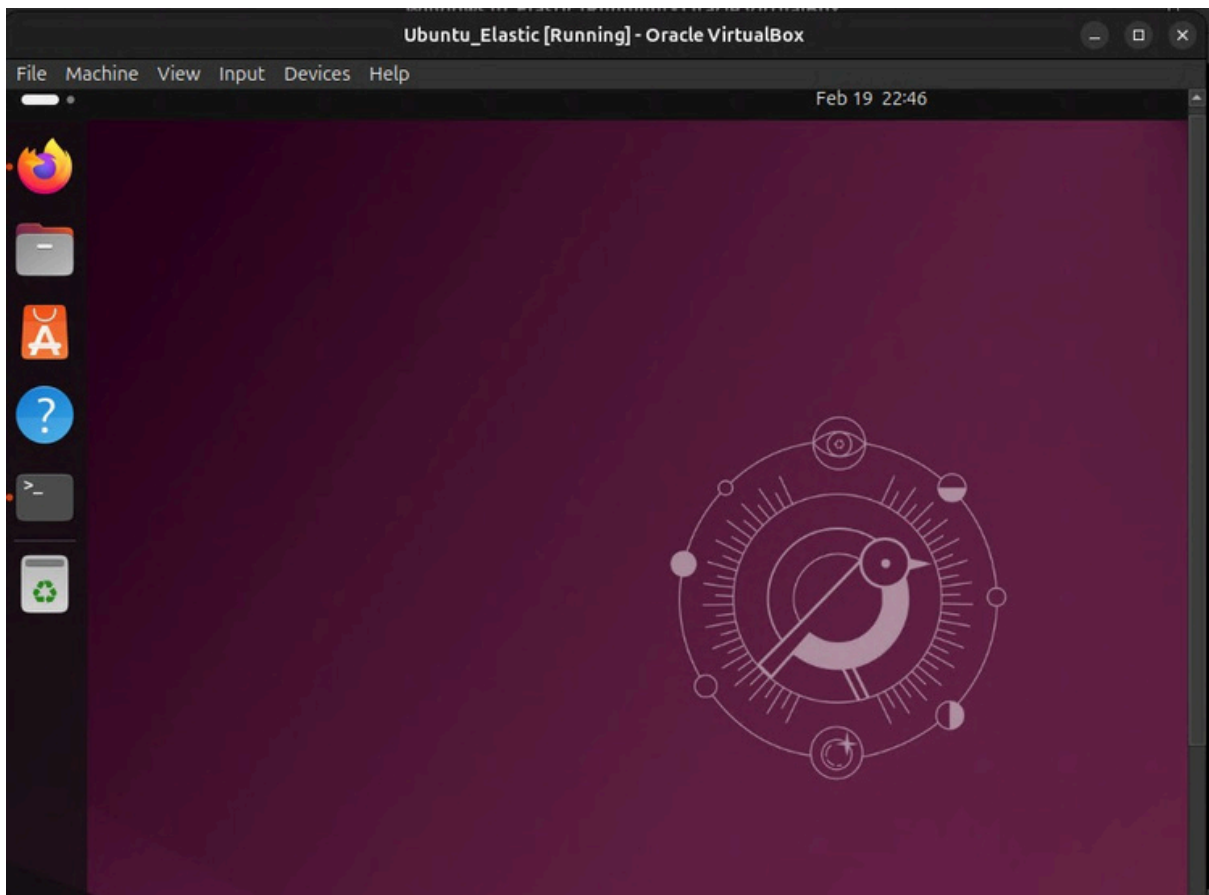
- Install Docker on an Ubuntu system.
- Install and set up Elastic.
- Verify connectivity using **ping**.
- Perform a network scan using **nmap**.
- Search for **ping** and **nmap** results in Elastic.

Tasks

Part 1: Setup Environment

You will need:

- A virtual machine (VM) running Ubuntu.
- Internet access to download required packages.



Ubuntu machine installed in a VM.

Part 2: Install Docker on Ubuntu

Follow the official installation guide for Docker:

[Docker Installation Guide](#)

```

tatiانا@tatiانا-VirtualBox:~$ # Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc \
https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
Hit:1 http://pt.archive.ubuntu.com/ubuntu oracular InRelease
Hit:2 http://pt.archive.ubuntu.com/ubuntu oracular-updates InRelease
Hit:3 http://pt.archive.ubuntu.com/ubuntu oracular-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu oracular-security InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

```

1. Update system packages:

```
sudo apt update && sudo apt upgrade -y
```

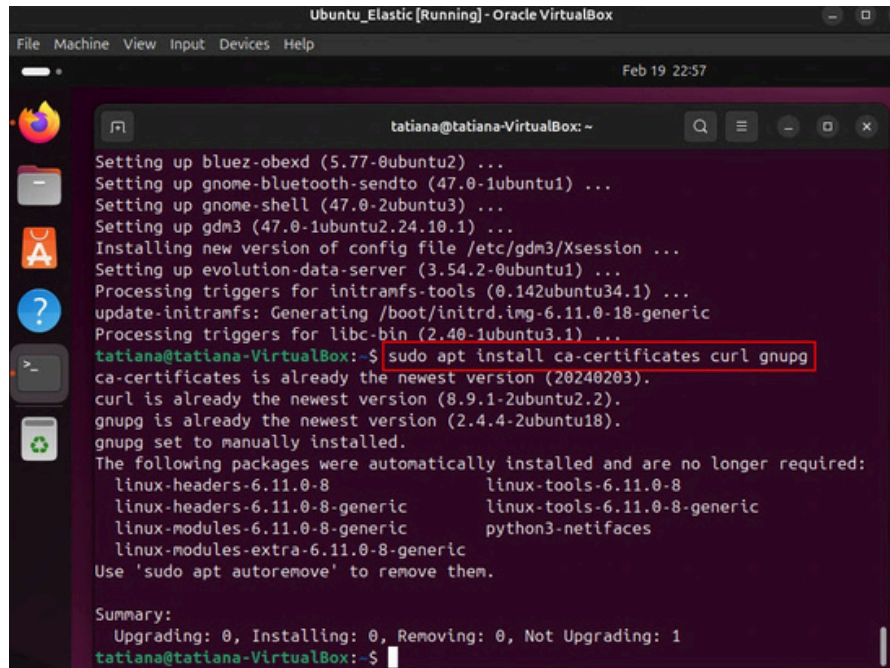
```

tatiانا@tatiانا-VirtualBox:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://pt.archive.ubuntu.com/ubuntu oracular InRelease
Hit:2 http://pt.archive.ubuntu.com/ubuntu oracular-updates InRelease
Hit:3 http://pt.archive.ubuntu.com/ubuntu oracular-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu oracular InRelease
Hit:5 http://security.ubuntu.com/ubuntu oracular-security InRelease
127 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
  linux-headers-6.11.0-8          linux-tools-6.11.0-8
  linux-headers-6.11.0-8-generic linux-tools-6.11.0-8-generic
  linux-modules-6.11.0-8-generic python3-netifaces
  linux-modules-extra-6.11.0-8-generic
Use 'sudo apt autoremove' to remove them.

```

2. Install prerequisite packages:

```
sudo apt install ca-certificates curl gnupg
```



```

Ubuntu_Elastic [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Feb 19 22:57

tatiana@tatiana-VirtualBox: ~
Setting up bluez-obexd (5.77-0ubuntu2) ...
Setting up gnome-bluetooth-sendto (47.0-1ubuntu1) ...
Setting up gnome-shell (47.0-2ubuntu3) ...
Setting up gdm3 (47.0-1ubuntu2.24.10.1) ...
Installing new version of config file /etc/gdm3/Xsession ...
Setting up evolution-data-server (3.54.2-0ubuntu1) ...
Processing triggers for initramfs-tools (0.142ubuntu34.1) ...
update-initramfs: Generating /boot/initrd.img-6.11.0-18-generic
Processing triggers for libc-bin (2.40-1ubuntu3.1) ...
tatiana@tatiana-VirtualBox:~$ sudo apt install ca-certificates curl gnupg
ca-certificates is already the newest version (20240203).
curl is already the newest version (8.9.1-2ubuntu2.2).
gnupg is already the newest version (2.4.4-2ubuntu18).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
  linux-headers-6.11.0-8      linux-tools-6.11.0-8
  linux-headers-6.11.0-8-generic  linux-tools-6.11.0-8-generic
  linux-modules-6.11.0-8-generic  python3-netifaces
  linux-modules-extra-6.11.0-8-generic
Use 'sudo apt autoremove' to remove them.

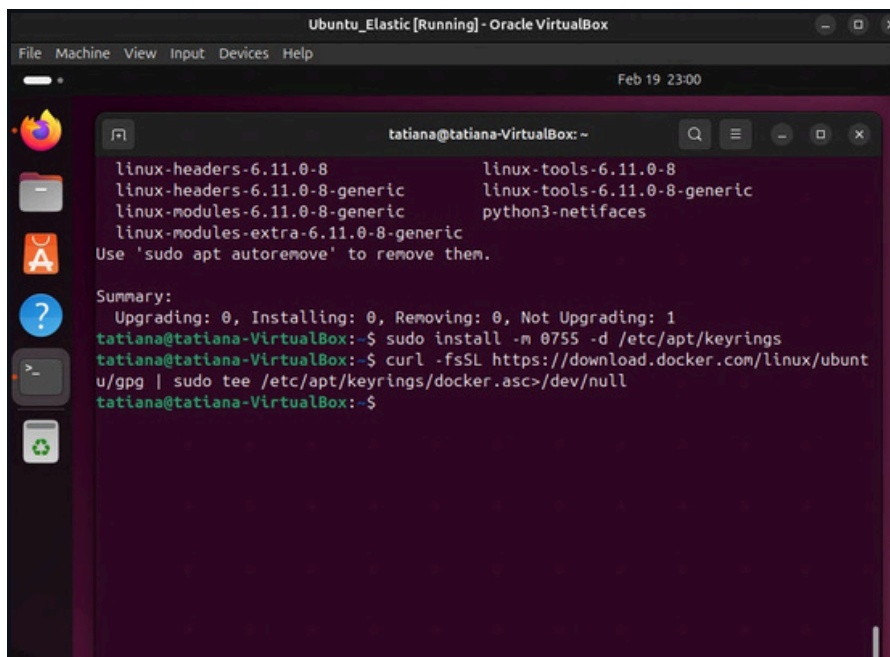
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1
tatiana@tatiana-VirtualBox:~$

```

3. Add Docker's official GPG key:

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo tee
/etc/apt/keyrings/docker.asc > /dev/null
```



```

Ubuntu_Elastic [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Feb 19 23:00

tatiana@tatiana-VirtualBox: ~
linux-headers-6.11.0-8      linux-tools-6.11.0-8
linux-headers-6.11.0-8-generic  linux-tools-6.11.0-8-generic
linux-modules-6.11.0-8-generic  python3-netifaces
linux-modules-extra-6.11.0-8-generic
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1
tatiana@tatiana-VirtualBox:~$ sudo install -m 0755 -d /etc/apt/keyrings
tatiana@tatiana-VirtualBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu
gpg | sudo tee /etc/apt/keyrings/docker.asc > /dev/null
tatiana@tatiana-VirtualBox:~$

```

4. Set up the repository: `echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null`


```

tutiana@tutiana-VirtualBox: ~
linux-headers-6.11.0-8          linux-tools-6.11.0-8
linux-headers-6.11.0-8-generic linux-tools-6.11.0-8-generic
linux-modules-6.11.0-8-generic python3-netifaces
linux-modules-extra-6.11.0-8-generic
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1
tutiana@tutiana-VirtualBox:~$ sudo install -m 0755 -d /etc/apt/keyrings
tutiana@tutiana-VirtualBox:~$ curl -fsSL https://download.docker.com/linux/ubuntu
u/gpg | sudo tee /etc/apt/keyrings/docker.asc > /dev/null
tutiana@tutiana-VirtualBox:~$ echo "deb [arch=$(dpkg --print-architecture) signe
d-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu $(ls
b_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/nul
l
tutiana@tutiana-VirtualBox:~$

```

5. Install Docker:

```
sudo apt update
```

```
sudo apt install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
```

```
docker-compose plugin
```

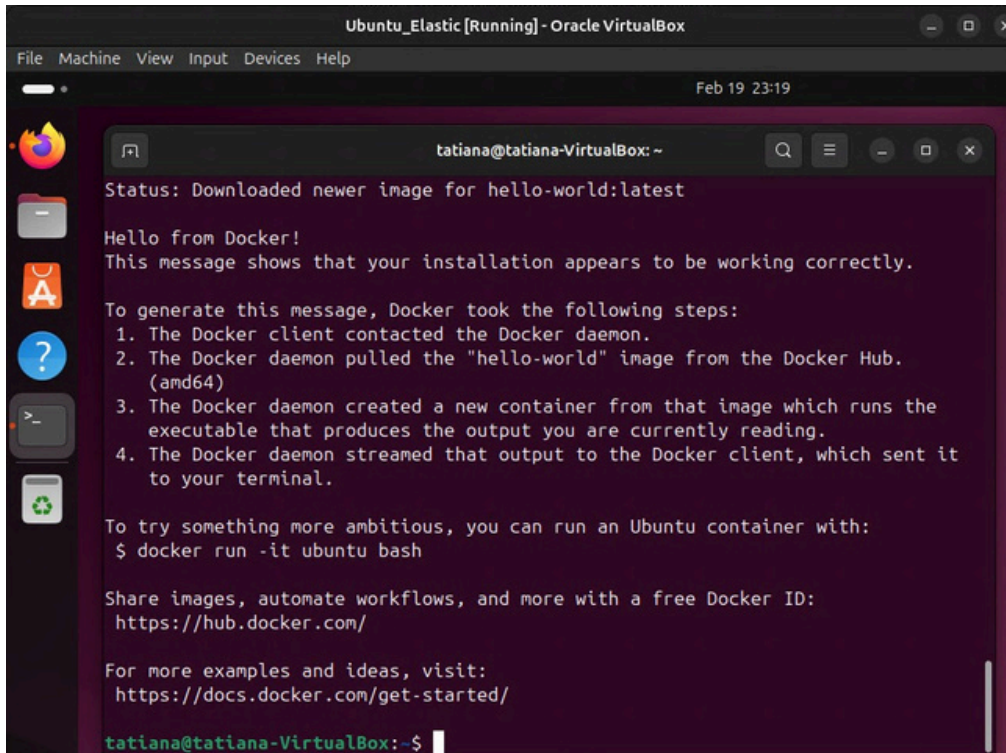
```

tutiana@tutiana-VirtualBox:~$ sudo apt update
Hit:1 http://pt.archive.ubuntu.com/ubuntu oracular InRelease
Hit:2 http://pt.archive.ubuntu.com/ubuntu oracular-updates InRelease
Hit:3 http://pt.archive.ubuntu.com/ubuntu oracular-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu oracular-security InRelease [126 kB]
Hit:5 https://download.docker.com/linux/ubuntu oracular InRelease
Get:6 http://security.ubuntu.com/ubuntu oracular-security/main amd64 Components
[8,996 B]
Get:7 http://security.ubuntu.com/ubuntu oracular-security/restricted amd64 Compone
nts [212 B]
Get:8 http://security.ubuntu.com/ubuntu oracular-security/universe amd64 Compone
nts [5,568 B]
Get:9 http://security.ubuntu.com/ubuntu oracular-security/multiverse amd64 Compone
nts [212 B]
Fetched 141 kB in 1s (107 kB/s)
1 package can be upgraded. Run 'apt list --upgradable' to see it.
tutiana@tutiana-VirtualBox:~$

```

6. Verify installation:

```
docker --version
```



The screenshot shows a terminal window titled "tatiana@tatiana-VirtualBox: ~" with a dark purple background. The output of the Docker installation is as follows:

```
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

tatiana@tatiana-VirtualBox:~$
```

Part 3: Install Elastic

1. Run the following command to download and install Elastic:

```
curl -fsSL https://elastic.co/start-local | sh
```

2. Follow the on-screen setup instructions.

```

Ubuntu_Elastic [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Feb 19 23:27

tatiana@tatiana-VirtualBox: ~/Elastic

Run Elasticsearch and Kibana for local testing

Do not use this script in a production environment

Setting up Elasticsearch and Kibana v8.17.2...

- Generated random passwords
- Created the elastic-start-local folder containing the files:
  - .env, with settings
  - docker-compose.yml, for Docker services
  - start/stop/uninstall commands
- Running docker compose up --wait

[+] Running 19/24
  :: kibana [#####] 390.6MB / 392.7MB Pulling 84.5s
  :: elasticsearch Pulling 84.5s
  :: kibana_settings [#####] 695.9MB / 702.6MB Pulling 84.5s

```

```

Ubuntu_Elastic [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Feb 19 23:31

tatiana@tatiana-VirtualBox: ~/Elastic

✓ elasticsearch Pulled 103.2s
✓ kibana_settings Pulled 103.3s
[+] Running 6/6
✓ Network elastic-start-local_default Created 0.4s
✓ Volume "elastic-start-local_dev-kibana" Created 0.0s
✓ Volume "elastic-start-local_dev-elasticsearch" Created 0.0s
✓ Container es-local-dev Healthy 134.1s
✓ Container kibana_settings Exited 132.6s
✓ Container kibana-local-dev Healthy 254.3s

Congrats, Elasticsearch and Kibana are installed and running in Docker!

Open your browser at http://localhost:5601

Username: elastic
Password: [REDACTED]

Elasticsearch API endpoint: http://localhost:9200
API key: UnEtTk1KVUJUaWNoNTZ5amt0a0E6cHJmLUJvV1JTRnU0VS1FbWJENEJYZw==

Learn more at https://github.com/elastic/start-local

tatiana@tatiana-VirtualBox:~/Elastic$

```

For further configuration, follow this guide:

[Elastic SIEM Lab Guide](#)

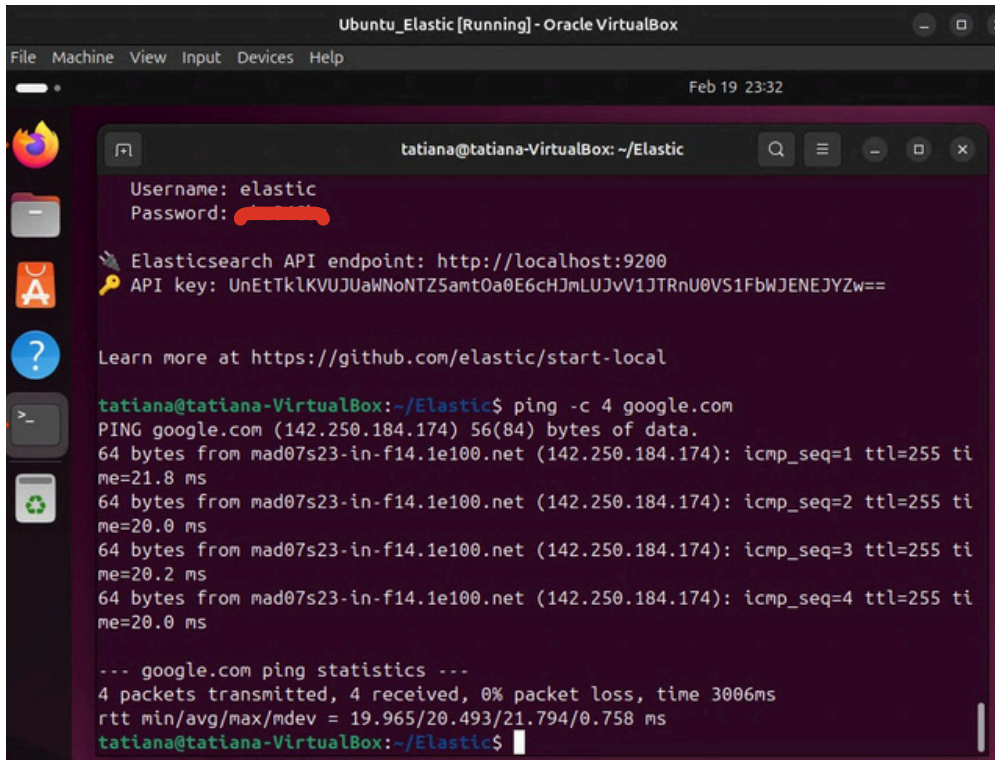
Part 4: Verify Connectivity

1. Ping Verification

To verify network connectivity, use the **ping** command:


```
ping -c 4 google.com
```

This ensures your system can reach external networks.



```
Ubuntu_Elastic [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Feb 19 23:32

tariana@tariana-VirtualBox: ~/Elastic
Username: elastic
Password: 
Elasticsearch API endpoint: http://localhost:9200
API key: UnEtTk1KVUJUaWNoNTZ5amt0a0E6cHJmLUJvV1JTRnU0VS1FbWJENEJYZw==
Learn more at https://github.com/elastic/start-local

tariana@tariana-VirtualBox:~/Elastic$ ping -c 4 google.com
PING google.com (142.250.184.174) 56(84) bytes of data.
64 bytes from mad07s23-in-f14.1e100.net (142.250.184.174): icmp_seq=1 ttl=255 time=21.8 ms
64 bytes from mad07s23-in-f14.1e100.net (142.250.184.174): icmp_seq=2 ttl=255 time=20.0 ms
64 bytes from mad07s23-in-f14.1e100.net (142.250.184.174): icmp_seq=3 ttl=255 time=20.2 ms
64 bytes from mad07s23-in-f14.1e100.net (142.250.184.174): icmp_seq=4 ttl=255 time=20.0 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 19.965/20.493/21.794/0.758 ms
tariana@tariana-VirtualBox:~/Elastic$
```

2. Nmap Scan

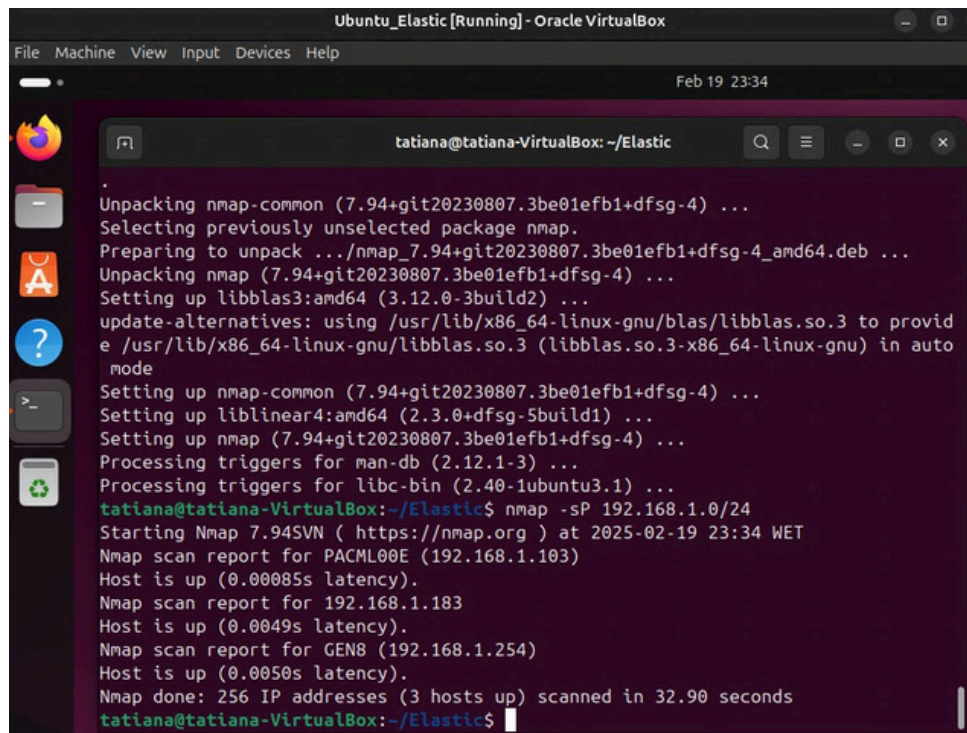
Install **nmap** if not already installed:

```
sudo apt install nmap -y
```

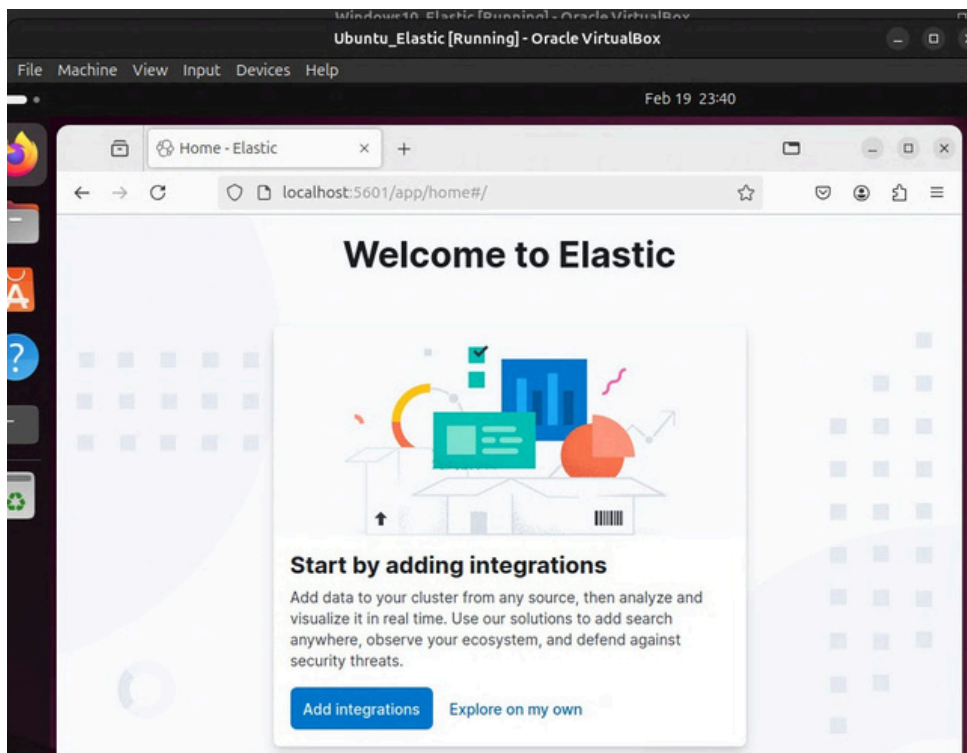
Perform a basic scan:

```
nmap -sP 192.168.1.0/24
```

This command scans all active hosts in your local network.

A terminal window titled 'Ubuntu_Elastic [Running] - Oracle VirtualBox' showing the installation and execution of nmap. The user 'tatiana' is at the prompt 'tatiana@tatiana-VirtualBox: ~/Elastic'. The terminal output shows the unpacking of nmap-common and nmap, setting up dependencies like libblas and liblinear, and then running 'nmap -sP 192.168.1.0/24'. The scan reports three hosts up: PACML00E (192.168.1.103), 192.168.1.183, and GEN8 (192.168.1.254).

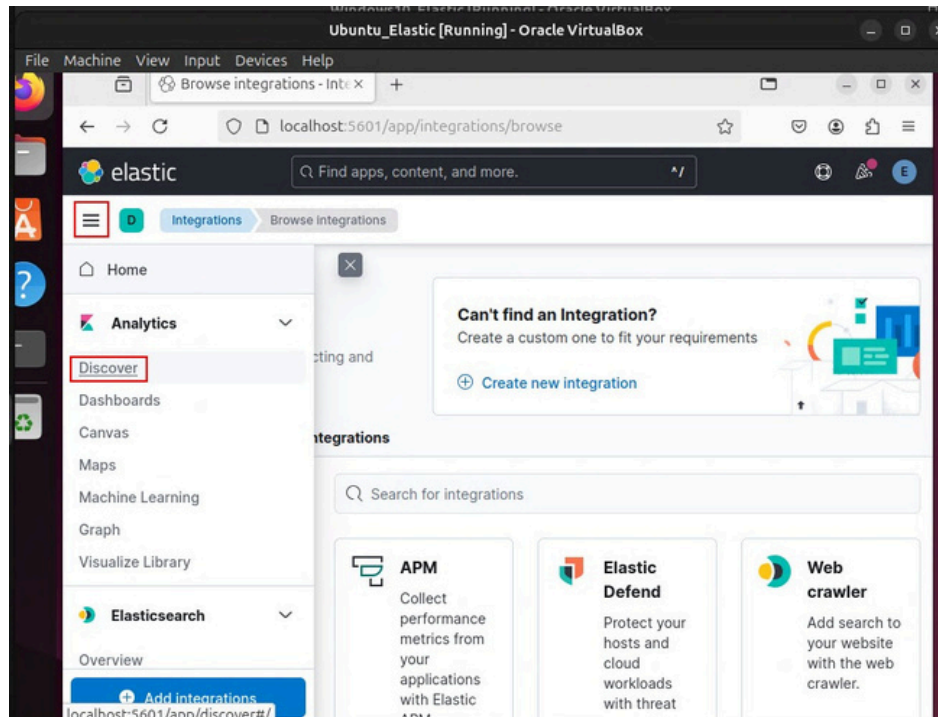
```
Unpacking nmap-common (7.94+git20230807.3be01efb1+dfsg-4) ...
Selecting previously unselected package nmap.
Preparing to unpack .../nmap_7.94+git20230807.3be01efb1+dfsg-4_amd64.deb ...
Unpacking nmap (7.94+git20230807.3be01efb1+dfsg-4) ...
Setting up libblas3:amd64 (3.12.0-3build2) ...
update-alternatives: using /usr/lib/x86_64-linux-gnu/blas/libblas.so.3 to provide
e /usr/lib/x86_64-linux-gnu/libblas.so.3 (libblas.so.3-x86_64-linux-gnu) in auto
mode
Setting up nmap-common (7.94+git20230807.3be01efb1+dfsg-4) ...
Setting up liblinear4:amd64 (2.3.0+dfsg-5build1) ...
Setting up nmap (7.94+git20230807.3be01efb1+dfsg-4) ...
Processing triggers for man-db (2.12.1-3) ...
Processing triggers for libc-bin (2.40-1ubuntu3.1) ...
tatiana@tatiana-VirtualBox:~/Elastic$ nmap -sP 192.168.1.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-02-19 23:34 WET
Nmap scan report for PACML00E (192.168.1.103)
Host is up (0.00085s latency).
Nmap scan report for 192.168.1.183
Host is up (0.0049s latency).
Nmap scan report for GEN8 (192.168.1.254)
Host is up (0.0050s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 32.90 seconds
tatiana@tatiana-VirtualBox:~/Elastic$
```



Part 5: Search for Ping and Nmap Results in Elastic

Once Elastic is set up, search for logs related to **ping** and **nmap**:

1. Navigate to the **Discover** tab.



2. Use the search bar to filter logs:

- For **ping**
- For **nmap**

3. Analyze the logs and take a screenshot of the results.

